

**REMARKS**

Applicant respectfully requests reconsideration and allowance of the subject application. Claims 13-22 and 38-47 are pending in this application.

**35 U.S.C. § 103**

Claims 13-22 and 38-47 stand rejected under 35 U.S.C. §103(a) as being unpatentable over U.S. Patent Application Publication No. 2001/0056504 to Kuznetsov (hereinafter "Kuznetsov") in view of The Component Object Model Specification, Version 0.9 (hereinafter "Comspec"). Applicant respectfully submits that claims 13-22 and 38-47 are not obvious over Kuznetsov in view of Comspec.

Kuznetsov is directed to data exchange and data transfer (see, Title and ¶ 1). As discussed in the Abstract of Kuznetsov, Kuznetsov describes a high level transformation method and apparatus for converting data formats in the context of network applications, among other places. A flexible transformation mechanism is provided that facilitates generation of translation machine code on the fly. Kuznetsov discusses conversions between different XML formats (see, ¶ 6), as well as XML to HTML and WAP formats (see, ¶ 7).

Comspec is Version 0.9 of the Component Object Model Specification. As discussed in Comspec at §1.3, pp. 14-15, "The Component Software Solution: OLE's COM", the Component Object Model is an object-based programming model designed to promote software interoperability; that is, to allow two or more applications or "components" to easily cooperate with one another, even if they were written by different vendors at different times, in different programming

languages, or if they are running on different machines running different operating systems. To support its interoperability features, COM defines and implements mechanisms that allow applications to connect to each other as *software objects*. A software object is a collection of related function (or intelligence) and the function's (or intelligence's) associated state. In other words, COM, like a traditional system service API, provides the operations through which a client of some service can connect to multiple providers of that service in a polymorphic fashion.

Furthermore, as discussed in Comspec at §1.3.2.2, p. 17, "COM's Standards Enable Object Interoperability", with COM, applications interact with each other and with the system through collections of function calls—also known as methods or member functions or requests—called *interfaces*. An "interface" in the COM sense is a strongly typed *contract* between software components to provide a relatively small but useful set of semantically related operations. An interface is an articulation of an expected behavior and expected responsibilities, and the semantic relation of interfaces gives programmers and designers a concrete entity to use when referring to the contract.

As discussed at MPEP §§ 2142 and 2143, to establish a *prima facie* case of obviousness, three basic criteria must be met. First, there must be some suggestion or motivation, either in the references themselves or in the knowledge generally available to one of ordinary skill in the art, to modify the reference or to combine reference teachings. Second, there must be a reasonable expectation of success. Finally, the prior art reference (or references when combined) must teach or suggest all the claim limitations. The teaching or suggestion to make the

claimed combination and the reasonable expectation of success must both be found in the prior art, and not based on applicant's disclosure. *In re Vaeck*, 947 F.2d 488, 20 USPQ2d 1438 (Fed. Cir. 1991).

Applicant respectfully submits that there is no suggestion or motivation to combine Kuznetsov and Comspec, and thus that no *prima facie* case of obviousness has been established. Kuznetsov, as discussed above, is directed to conversions between different XML formats, as well as conversions from XML to HTML and WAP formats. Kuznetsov is thus directed to simply converting between different data formats. Comspec, as discussed above, is directed to the Component Object Model, which is an object-based programming model designed to promote software interoperability. The Component Object Model as discussed in Comspec, however, is much more than simply a new data format. For example, as discussed above COM includes the mechanisms that allow applications to connect to each other as software objects and interfaces that allow applications to interact with each other. There is no suggestion or motivation, either in the references themselves or in the knowledge generally available to one of ordinary skill in the art, to combine the data format conversion of Kuznetsov with some other kind of conversion that is not conversion to simply another data format.

In the August 25, 2005 Office Action at ¶ 4, p. 13, it was asserted that:

In this case, the modification would have been obvious because one of ordinary skill in the art would have wanted the flexibility of converting a recent data encoding format, such as XML, into the format of an existing technology, such as COM, (Kuznetsov, ¶ 7:13-16).

This cited portion of Kuznetsov reads as follows:

As the diversity of web-connected devices grows, so grows the need to provide dynamic conversion, such as XML-to-HTML and XML-to-WAP, for e-business applications.

Applicant respectfully submits that this cited portion discusses converting to different data formats, not converting to something else that is not simply another data format. The Component Object Model, as discussed in Comspec, is not simply another data format. For example, as discussed above COM includes the mechanisms that allow applications to connect to each other as software objects and interfaces that allow applications to interact with each other. As there is no discussion or mention of converting to something else that is not simply another data format, Applicant respectfully submits that Kuznetsov does not provide any suggestion or motivation to combine Kuznetsov with Comspec.

Furthermore, assuming for the sake of argument that Kuznetsov and Comspec were combined, Applicant respectfully submits that the combination does not disclose or suggest the elements of claim 13. Claim 13 recites:

One or more computer readable media having stored thereon a plurality of instructions that, when executed by a transformation engine, causes the transformation engine to:

access a plurality of constructs in an application programming interface description, wherein the description is written in an extensible markup language (XML) format; and

transform each of the plurality of constructs into code for a component object module (COM) application programming interface header file.

Applicant respectfully submits that there is no disclosure or suggestion in Kuznetsov or Comspec to transform each of a plurality of constructs into code for a COM application programming interface header file as recited in claim 13.

In the August 25, 2005 Office Action at ¶ 3, p. 3, it was acknowledged that Kuznetsov doesn't explicitly disclose translation into a COM application programming interface header file, but asserted that Comspec discloses COM application programming interface. As discussed above, COM includes the mechanisms that allow applications to connect to each other as software objects and interfaces that allow applications to interact with each other. An XML document does not have such mechanisms and interfaces, and thus cannot be simply converted into COM. There is no discussion or suggestion in Kuznetsov or Comspec of how such an XML document could be converted into COM because such mechanisms and interfaces would need to be generated and there is no discussion of how to generate such mechanisms and interfaces from XML. Without any such discussion or suggestion, Applicant respectfully submits that Kuznetsov in view of Comspec cannot disclose or suggest to transform each of the plurality of constructs into code for a component object module (COM) application programming interface header file as recited in claim 13.

Accordingly, for at least these reasons, Applicant respectfully submits that claim 13 is allowable over Kuznetsov in view of Comspec.

With respect to claims 14 and 16-22, given that claims 14 and 16-22 depend from claim 13, Applicant respectfully submits that claims 14 and 16-22 are likewise allowable over Kuznetsov in view of Comspec for at least the reasons discussed above with respect to claim 13.

With respect to claim 15, claim 15 depends from claim 13 and Applicant respectfully submits that claim 15 is allowable over Kuznetsov in view of

Comspec for at least the reasons discussed above with respect to claim 13.

Furthermore, claim 15 recites:

One or more computer readable media as recited in claim 13,  
wherein the plurality of instructions include instructions to:

check whether a declare enumeration construct is to be  
transformed into a series of manifest constants or into a component  
object model enumeration declaration; and

transform the enumeration construct into either the series of  
manifest constants or the component object model enumeration  
declaration based on the checking.

Applicant respectfully submits that no such check and transformation is disclosed  
or suggested by Kuznetsov in view of Comspec.

In the August 25, 2005 Office Action at ¶ 3, p. 4-5, Comspec is relied on as  
disclosing this check and transformation of claim 15. More specifically, it was  
asserted that:

However, Comspec, in an analogous environment, discloses  
that the plurality of instructions include instructions to:

- check whether a declare enumeration construct is to be  
transformed into a series of manifest constants or into a component  
object model enumeration declaration (p. 8:30, "this is a manifest  
constant defined in the header files", and p. 7:1, "enumeration  
(declaration)", and to perform transformation between XML and  
COM, the transformation engine maps constructs, constants and  
declarations in XML to the corresponding constructs, constants and  
declarations in COM),

Applicant respectfully disagrees. The mere mention of a manifest constant and an  
enumeration in Comspec does not provide any disclosure or suggestion to check  
whether a declare enumeration construct is to be transformed into a series of  
manifest constants or into a component object model enumeration declaration.  
Although a manifest constant and an enumeration are mentioned in Comspec,  
there is no discussion or mention of any check as to which of the manifest constant

and the enumeration a declare enumeration construct is to be transformed into. Without any such discussion or mention, Applicant respectfully submits that Comspec cannot disclose or suggest to check whether a declare enumeration construct is to be transformed into a series of manifest constants or into a component object model enumeration declaration as recited in claim 15.

With respect to Kuznetsov, Kuznetsov is not cited as curing, and does not cure, these deficiencies of Comspec. Accordingly, for at least these reasons, Applicant respectfully submits that claim 15 is allowable over Kuznetsov in view of Comspec.

With respect to claim 38, Applicant respectfully submits that, as discussed above with respect to claim 13, it would not have been obvious to combine Kuznetsov and Comspec. Accordingly, for at least these reasons, Applicant respectfully submits that claim 38 is allowable over Kuznetsov in view of Comspec.

Given that claims 39 and 41-47 depend from claim 38, Applicant respectfully submits that claims 39 and 41-47 are likewise allowable over Kuznetsov in view of Comspec for at least the reasons discussed above with respect to claim 38.

With respect to claim 40, claim 40 depends from claim 38 and Applicant respectfully submits that claim 40 is allowable over Kuznetsov in view of Comspec for at least the reasons discussed above with respect to claim 38. Furthermore, similar to the discussion above regarding claim 15, Applicant respectfully submits that Kuznetsov in view of Comspec does not disclose or suggest an enumeration flag attribute that identifies whether the plurality of

declare enumeration member constructs are to be transformed into a series of manifest constants or transformed into a component object model enumeration declaration as recited in claim 40. Accordingly, for at least these reasons, Applicant respectfully submits that claim 40 is allowable over Kuznetsov in view of Comspec.

Applicant respectfully requests that the §103 rejections be withdrawn.

### Conclusion

Claims 13-22 and 38-47 are in condition for allowance. Applicant respectfully requests reconsideration and issuance of the subject application. Should any matter in this case remain unresolved, the undersigned attorney respectfully requests a telephone conference with the Examiner to resolve any such outstanding matter.

Respectfully Submitted,

Date: 10/25/05

By: 

Allan T. Sponseller  
Reg. No. 38,318  
(509) 324-9256